

15-112 Fall 2022 Lecture 3

Quiz 2B

35 minutes

Name: _____

Andrew ID: _____@andrew.cmu.edu

Section: _____

- You may not use any books, notes, or electronic devices during this quiz.
- You may not ask questions about the quiz except for language clarifications.
- Show your work on the quiz (not scratch paper) to receive credit.
- If you use scratch paper, you must submit it with your andrew id on it, and we will ignore it.
- All code samples run without crashing unless we state otherwise. Assume any imports are already included as required.
- Do not use these topics: sets/dictionaries and recursion.
- You may use `almostEqual()` and `rounded()` without writing them. You must write everything else.

Do not write below here

Question	Points	Score
1. CT	45	
2. FR: Dot around dot	55	
3. Bonus	5 (bonus)	
TOTAL	100	

1. CT [50 pts]

Indicate what these print. Place your answers (and nothing else) in the box next to each block of code.

```
def ct1(L, M):  
    N = L  
    L += [2]  
    M += [1]  
    N = N + [0]  
    return L, M, N  
L = [3]  
L, M, N = ct1(L, L+[4])  
print(L, M, N)
```

```
def ct2(L):  
    M = sorted(L)  
    N = list(reversed(M))  
    n = L.pop(0) * 10  
    L.remove(3)  
    L.sort()  
    L.append(n)  
    L.extend(L[:1])  
    return (L.index(1), M.index(1), N.index(1))  
L = [3,2,1,2,3]  
M = ct2(L)  
print(L, M, type(M) == tuple)
```

```
def ct3(s):  
    L = [int(v) for v in s.split(',')]  
    M = [  
        L > L[:-1],  
        L[-1:]*L[-1],  
        sum(L) - min(L)  
    ]  
    N = [str(v) for v in M]  
    return '-'.join(N)  
print(ct3('12,4,3'))
```



2. Free Response: Dot around Dot [50 pts]

Write an app such that:

1. At first, a cyan dot of radius 20 is drawn so it is centered vertically but its left edge is at the left edge of the canvas. Also, a smaller black dot of radius 5 is drawn at "12 o'clock" on the cyan circle -- that is, so the center of the black dot is at the top of the cyan circle.
2. Every $1/20$ th second, the black dot moves around the cyan dot, with its center always on the edge (or perimeter) of the cyan dot, changing its position by 10 degrees. At first this motion is clockwise, but each time the user presses any key, the motion toggles between clockwise and counter-clockwise.
3. Every $1/4$ second, the cyan dot moves 10 pixels to the right, until it hits the right edge of the canvas, then it moves 10 pixels to the left until it hits the left edge of the canvas, at which point the cyan dot no longer moves.
 - Hint: you should set `app.stepsPerSecond` to 20, but only have the cyan dot move on some but not all calls to `onStep`.
4. Note that the black dot always moves around the cyan dot, even as the cyan dot itself is moving. Also, the black dot continues to move, even after the cyan dot stops moving.

Hint: to solve this, you will need to write `onAppStart`, `redrawAll`, `onStep`, and `onKeyPress`. Also, you may assume the canvas is 400x400, and that all necessary imports are already included for you.

This page is blank (for your Dot around Dot solutions, if needed).

3. Bonus [5 pts]

Indicate what these print. Place your answers (and nothing else) in the box next to each block of code.

```
def bonusCt1(L):  
    while sum(L) < L[0]**L[1]:  
        L += L * sum(L)  
    return L  
print(len(bonusCt1([3,5])))
```

```
def bonusCt2(L, n):  
    while L:  
        L, n = L[:-1], 10*n+L[-1]%10  
    L = list(str(n))  
    return [int('2' + ''.join(L[:i]))  
            for i in range(0, len(L), 2)]  
print(bonusCt2([1,23,45,67],9))
```